

# Android OS を搭載した Raspberry Pi4 クラスターの MPI 性能評価

横山 尚弥<sup>†</sup>  
北陸先端科学技術大学院大学<sup>†</sup>

井口 寧<sup>‡</sup>  
北陸先端科学技術大学院大学<sup>‡</sup>

## 1 はじめに

近年、サイネージ端末のような小型固定端末の性能向上が著しく、ネットワークを介して複数の小型固定端末上で並列処理を行うことでデータセンターのサーバで動かすような大きな計算量の処理を行うことができる。

本研究では Android OS を搭載した複数の Raspberry Pi4 [1] 間でネットワークを介して並列処理を行うようなシステムでも、プロセス数の増加に応じて並列性能が向上することを示す。

## 2 先行研究

杉山らの先行研究 [2] や飯村らの先行研究 [3] では Android OS を搭載した Raspberry Pi に対して SPEC MPI ベンチマークソフトを用いてクラスターの並列処理性能を計測し、無線環境にて NAS Parallel Benchmarks(以下 NPB)[4] の EP テスト、Himeno benchmark、手書き文字の学習アプリケーション上では筐体を跨いだ並列処理でも性能が向上する事を示した。

表 1 通信環境

環境名	有線 LAN 環境	無線 wifi 環境
通信媒体	スイッチングハブ	wifi ルーター
通信環境	1000BASE-T	IEEE 802.11ax

MPI Performance Evaluation of Raspberry Pi4 Cluster with Android OS

<sup>†</sup> Naoya Yokoyama, Japan Advanced Institute of Science and Technology

<sup>‡</sup> Yasushi Inoguchi, Japan Advanced Institute of Science and Technology

表 2 Raspberry Pi4 の仕様

CPU	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
LAN カード	Gigabit Ethernet
メモリ	8GB LPDDR4 memory
ディスク	32GB SD カード
無線 LAN	5.0 GHz IEEE 802.11ax wireless

表 3 ソフトウェア情報

OS	Android 11
NAS Parallel Benchmarks	3.4.2
GCC, GCC FORTRAN	10.2

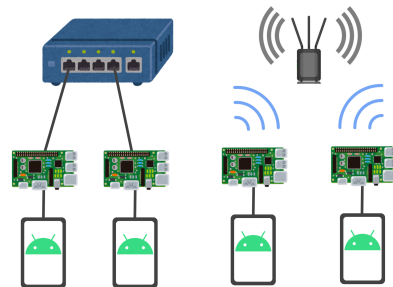


図 1 左: 有線 LAN 環境、右: 無線 wifi 環境

## 3 実装

### 3.1 環境

本実装に用いたハードウェア環境を表 2 に、その構成を図 1 に示す。また、通信は有線および無線で行ったが、その詳細を表 1 に示す。更に、ソフトウェア環境は表 3 に示す。

### 3.2 MPI 環境の実装

本実験では Android OS を積載した Raspberry Pi 上にて MPI 環境を構築した。通常 Android OS は特権ユーザで無ければ internal

storage 上で実行する事ができない。そこで本実験では別途ファイルシステムを external storage 上で構築し、そのファイルシステムに MPI プログラムを実行する上で必要なライブラリをインストールを行い terminal emulator を通じて実行する事で一般ユーザ権限を用いて MPI アプリケーションを実行することを可能とした。

### 3.3 実行した並列処理について

本実験での MPI 並列処理について。まず Linux 上で MPI 並列処理を起動するため 1 つの mpirun プロセスをノードごとに起動する。その後、mpirun はそれぞれのノード毎に MPI 並列処理を行うプロセスを管理し実行を行う。

## 4 評価

本実験では並列処理の性能を測るために NPB と呼ばれる並列コンピュータ上での実行性能を評価するベンチマークソフトを使用した。NPB の各テストについては、クラスターサイズに応じて、計算量の順に応じてクラスを選択する事が出来る。本実験では十分にパフォーマンス測定が出来る A クラスを選択し、ネットワーク負荷の比較を行うのに適している EP、CG、MG の 3 つのテストを行った。

EP は通信は殆ど使用せず、多数のガウス分布に従う擬似乱数を用いた試験を行う。CG は不規則な長距離通信を用いて、正值対称な疎行列の最小固有値を共役勾配法によって求める試験を行う。MG は短距離および長距離の通信を用いて、三次元ポアソン方程式をマルチグリッド法によって求めるテストである。

実験結果については表 4 に記した。プロセス数およびノード数の表記は  $x(n)$  のように表し、 $x$  はプロセス数、 $n$  はノード数とする。例えば 4(1) は 4 プロセスの並列計算を 1 ノードで実験を行ったという意味である。また本実験で使用した Raspberry Pi 4 は 4 コア CPU であり、ノードが並列処理に加わった場合、その 4 コア全てに対して計算の割り当てを行った。

表 4 から、先行研究 [2] でも無線環境だけの測定であれば EP のみノードを跨いだ場合でも並列性能を上げた事がわかっており、本研究でも同様の結果となった。一方、本研究では MG や CG など多くのノード間通信を行う試験でも、本研究の有線環境であれば、筐体を跨いだ場合でも合計プロセス数をあげていくほど並列処理の性能は向上することを示せた。

表 4 無線/有線環境における実験結果 (単位は秒)

プロセス数 (ノード数)	無線環境			有線環境		
	EP	MG	CG	EP	MG	CG
4(1)	6.95	5.80	3.68	6.95	5.80	3.68
8(2)	3.62	10.66	12.60	3.58	3.24	2.54
16(4)	1.99	9.79	15.54	1.84	2.01	2.30

## 5 結論

本報告では Raspberry Pi4 の筐体間での並列処理評価を目的として、NPB による試験を行った。その結果筐体を跨いでも Raspberry Pi4 を用いた有線構成であれば NPB の MG や CG 程度の通信を要しても十分に並列性能向上が見込めた。今後の課題としては本実験環境においてこういった通信でオーバーヘッドが生じるかより多くの試験により評価する事である。

## 参考文献

- [1] <https://www.raspberrypi.org/products/>
- [2] ”杉山 裕紀、大津 金光、大川 猛、横田 隆史、無線接続型 Android クラスタにおける MPI 並列プログラムの性能評価”、電子情報通信学会、pp.1~6、2018
- [3] ”飯村 琢泉、新里 将大、大津 金光、横田 隆史、Android クラスタにおける SPEC MPI を用いた並列処理性能の初期評価”、情報処理学会第 82 回全国大会、pp.1-91~1-92、2020
- [4] <https://www.nas.nasa.gov/software/npb.html>